Amendment to the Claims:

This listing of claims replaces all prior versions, and listings, of claims in the application:

1. (Previously Presented) An apparatus comprising:

a cache of trace information associated with a plurality of traces, each of the plurality of traces including information indicative of interdependent instructions, which interdependent instructions include at least an associated instruction and a criterion instruction that is part of a program sequence and which is data dependent on said associated instruction; and

one or more processors that speculatively execute interdependent instructions associated with a first trace of the plurality of traces as a result of detecting a first triggering condition corresponding to the first trace.

2. (Previously Presented) The apparatus of claim 1 wherein the trace information comprises a directed acyclic graph.

3. (Previously Presented) The apparatus of claim 1 wherein the trace information includes pointers to the interdependent instructions.

4. (Cancelled)

5. (Original) The apparatus of claim 1 wherein the interdependent instructions include the criterion instruction and instructions preceding the criterion instruction in the program sequence.

6. (Previously Presented) The apparatus of claim 1 wherein the interdependent instructions are classified into subslice types, the trace information including a pointer to each subslice that is formed by each type of the interdependent instructions.

7. (Original) The apparatus of claim 6 wherein each subslice is stored as dependent pieces.

8. (Previously Presented) The apparatus of claim 1 wherein the first triggering condition comprises a triggering instruction in the program sequence.

9. (Previously Presented) The apparatus of claim 8 wherein the first triggering condition is based on evaluation of an architectural state.

10. (Previously Presented) The apparatus of claim 8 wherein the first triggering condition is based on evaluation of a micro-architectural state.

11. (Previously Presented) The apparatus of claim 1 wherein the one or more processors further determine a confidence metric of trace information associated with a specific trace, and wherein the confidence metric is indicative of a likelihood of producing a correct result from executing the specific trace.

12. (Previously Presented) The apparatus of claim 11 wherein the confidence metric indicates whether or not the specific trace should be replaced by a new trace storing information about different instructions.

13. (Previously Presented) The apparatus of claim 11 wherein the confidence metric indicates whether or not the specific trace should be rebuilt using new information about a criterion instruction associated with the specific trace.

14. (Previously Presented) The apparatus of claim 11 further comprising a counter having a counter value that indicates the number of times the specific trace has been

executed, the counter value, when exceeding a frequency threshold associated with the specific trace, triggering the specific trace to be rebuilt.

15. (Previously Presented) The apparatus of claim 1 wherein the plurality of traces includes a second trace and a third trace, wherein the second and third traces are independent of each other and adjacent in the program sequence, and further comprising grouping the second trace and the third trace into a very-long-instruction-word.

16. (Previously Presented) The apparatus of claim 1 wherein traces of the plurality of traces that are data dependent of each other are chained together for serial executions.

17. (Previously Presented) The apparatus of claim 1 further comprising an instruction pointer that indexes the trace information, the instruction pointer pointing to a first instruction or a last instruction of the interdependent instructions.

18. (Previously Presented) The apparatus of claim 1 further comprising:

a main pipeline executing the program sequence; and at least one secondary pipeline disjoint from the main pipeline configured to speculatively execute the interdependent instructions associated with the first trace.

19. (Previously Presented) The apparatus of claim 1 wherein the interdependent instructions associated with the first trace are executed by a secondary thread on a pipeline, and the program sequence is executed by a main thread on the same pipeline.

20. (Previously Presented) A method comprising:

identifying a criterion instruction capable of incurring a potential latency in a program sequence, the potential latency associated with at least one of fetching an instruction associated with an incorrectly chosen branch of the criterion instruction and executing the criterion instruction and incurring a cache miss;

identifying a pre-selected number of initial candidate instructions preceding the criterion instruction in the program sequence;

determining which of the initial candidate instructions are associated instructions, wherein an outcome of the criterion instruction depends on the results of the associated instructions; and

storing information indicative of the associated instructions and the criterion instruction as trace information in a trace cache.

21. (Previously Presented) The method of claim 20 wherein the trace information is in a form of a directed acyclic graph.

22. (Previously Presented) The method of claim 20 further comprising determining at least one of whether a length of the potential latency, exceeds a predetermined time threshold, whether a frequency of the potential latency exceeds a predetermined recurrence threshold, and whether a variance of the potential latency exceeds a variance threshold.

23. (Previously Presented) The method of claim 20 further comprising dynamically identifying the criterion instruction based on information derived from at least one previous execution of the program sequence.

24. (Previously Presented) The method of claim 20 further comprising capturing the criterion instruction and the associated instructions by copying the criterion instruction and the associated instructions in a buffer.

25. (Previously Presented) The method of claim 20 further comprising locating existing trace information in the trace cache before storing the trace information, the existing trace information and the trace information to be stored having at least one of the same first instruction or the same last instruction.

26. (Previously Presented) The method of claim 20 further comprising rebuilding the trace information after a duration of time interval that grows each time the trace information is rebuilt until the duration reaches a predetermined time limit.

27. (Previously Presented) The method of claim 20 further comprising storing, in an array, the information indicative of the associated instructions and the criterion instruction as trace information.

28. (Previously Presented) The method of claim 27 wherein the array further includes a subslice type for each of the

corresponding criterion instruction and associated instructions, the subslice type being a result of classifying the criterion instruction and associated instructions.

29. (Currently Amended) A computer program residing on a computer readable medium comprising instructions for causing one or machines ~~to~~:

identify a criterion instruction incurring a potential latency in a program sequence, the potential latency associated with at least one of fetching an instruction associated with an incorrectly chosen branch of the criterion instruction and executing the criterion instruction and incurring a cache miss;

identify a pre-selected number of initial candidate instructions preceding the criterion instruction in the program sequence;

determine which of the initial candidate instructions are associated instructions, wherein an outcome of the criterion instruction depends on the results of the associated instructions; and

~~storeinformation~~ store information indicative of the associated instructions and the criterion instruction as trace information in a trace cache.

30. (Previously Presented) The computer program of claim 29 wherein an analysis window defined in the computer program causes the computer to capture the criterion instruction and initial candidate instructions.

31. (Original) The computer program of claim 29 wherein the computer identifies the criterion instruction by profiling the program sequence.

32. (Previously Presented) The apparatus as in claim 1, wherein there are multiple associated instructions associated with a criterion instruction of the first trace, and said processor forms first trace information using a first of said associated instructions, and second trace information using a second of said associated instructions.

33. (Previously Presented) The apparatus as in claim 1, wherein said criterion instructions are capable of incurring a potential latency in the program sequence, the potential latency associated with at least one of fetching an instruction associated with an incorrectly chosen branch of the criterion instruction and executing the criterion instruction and incurring a cache miss.

34. (Previously Presented) The apparatus as in claim 1, wherein the criterion instructions include template data for an instruction form, and the associated instructions include information that assigns values of register information within the criterion instructions.

35. (Previously Presented) The method as in claim 20, further comprising associating multiple different associated instructions with said criterion instruction, and wherein said storing comprises storing first trace information indicative of a first of said associated instructions, and second trace information indicative of a second of said associated instructions.

36. (Previously Presented) The method as in claim 20, wherein said potential latency includes a time to unload the instruction associated with the incorrectly chosen branch of the criterion instruction from a pipeline.

37. (Previously Presented) The method as in claim 20, wherein the criterion instructions include data for an instruction form, and said storing comprising associating information from said additional instructions as register information within the criterion instructions.

38. (Previously Presented) A program as in claim 29, further comprising associating multiple different associated instructions with said criterion instruction, and wherein said instructions cause the one or machines to store first trace information using a first of said associated instructions, and further to store second trace information using a second of said associated instructions.

39. (Previously Presented) A program as in claim 29, wherein said potential latency includes a time to unload the instruction associated with the incorrectly chosen branch of the criterion instruction from a pipeline.

40. (Cancelled)